HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10992578-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Cherkasova, et al **Confirmation No.: 5563**

Application No.: 09/502,718 **Examiner:** Nguyen, Chau

Filing Date: 2/11/00 **Group Art Unit:** 2142

Title: A Method for Allocating Web Sites on a Web Server Cluster Based on Balancing Memory and Load Requirements

COMMISSIONER FOR PATENTS
Washington, D.C. 20231

TRANSMITTAL LETTER FOR RESPONSE/AMENDMENT

Sir:

Transmitted herewith is/are the following in the above-identified application:

( ) Response/Amendment ( ) Petition to extend time to respond
( ) New fee as calculated below ( ) Supplemental Declaration
( ) No additional fee (Address envelope to "Box Non-Fee Amendments")
(X) Other: Appeal Brief (fee $ 320.00)

CLAIMS AS AMENDED BY OTHER THAN A SMALL ENTITY						
(1) FOR	(2) CLAIMS REMAINING AFTER AMENDMENT	(3) NUMBER EXTRA	(4) HIGHEST NUMBER PREVIOUSLY PAID FOR	(5) PRESENT EXTRA	(6) RATE	(7) ADDITIONAL FEES
TOTAL CLAIMS		MINUS		= 0	X $18	$ 0
INDEP. CLAIMS		MINUS		= 0	X $84	$ 0
[ ] FIRST PRESENTATION OF A MULTIPLE DEPENDENT CLAIM					+ $280	$ 0
EXTENSION FEE	1ST MONTH $110.00 [ ]	2ND MONTH $410.00 [ ]	3RD MONTH $930.00 [ ]	4TH MONTH $1450.00 [ ]		$ 0
					OTHER FEES	$ 320
					TOTAL ADDITIONAL FEE FOR THIS AMENDMENT	$ 320

Charge $ 320 to Deposit Account 08-2025. At any time during the pendency of this application, please charge any fees required or credit any overpayment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16, 1.17, 1.19, 1.20 and 1.21. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Cherkasova, et al

By _____

Calvin B. Ward

Attorney/Agent for Applicant(s)
Reg. No. 30,896
Date: 8/7/03

Telephone No.: (925) 855-0413

(X) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number (703) 746-7238 on 8/7/03
Number of pages: 46

Typed Name: Calvin B. Ward

Signature: _____

Rev 02/03 (TnAmdFax)

- Attach as First Page to Transmitted Papers -

SC
#8-10
10/8/03

PATENT APPLICATION
Attorney Docket: 10092578-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF APPEALS

Applicant:	Cherkasova, et al.
Serial No.:	09/502,718
Filed:	2/11/2000
For:	A Method for Allocating Web Sites on a Web Server Cluster Based on Balancing Memory and Load Requirements
Group Art Unit:	2142
Examiner:	Nguyen, Chault

BRIEF FOR APPELLANT

Hon. Commissioner of Patents
and Trademarks
Washington, D.C. 20231

Sir:

This is an appeal from the decision of the Primary Examiner dated 4/8/03, finally rejecting Claims 1-12 in the above-identified patent application.

I. REAL PARTY IN INTEREST

The real party in interest is Hewlett Packard Company, having an address as shown below.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to appellant, the appellant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in this pending appeal.

III. STATUS OF THE CLAIMS

10/08/2003 SCHAPMAN 00000001 082025 09502718
1 FC:1402 320.00 DA
2 FC:1251 110.00 DA

Claims 1-12 are currently pending in the above-identified patent application. In the Office Action dated 4/8/2003, the Examiner rejected Claims 1-12 and indicated that the action was final.

IV. STATUS OF AMENDMENTS

A response was to this office action was filed on 6/12/03. In an advisory action dated 6/25/2003, the Examiner indicated that the response would be entered on filing an appeal.

V. SUMMARY OF THE INVENTION

Referring to Figure 1 and the specification starting at line 22 of page 7, the present invention as claimed in the appealed claims is a method for operating a server cluster 13 that includes N server nodes 14 that service client requests from users 11. Each client request is directed to one of a plurality of sites hosted on the server cluster 14. Each site is identified by a domain name(page 6, line 16), and each server node is identified by an address(page 7, line 26) on a network connecting the clients to the server nodes. The computational resources required to service the requests to each of the sites over a first time period are measured and used to group the sites into N groups. Each group is assigned to a corresponding one of the server nodes 14. The groups are chosen such that, for each pair of groups, the difference in the sum of the measured computational resources is within a first predetermined error value(Page 16, starting at line 21). Configuration information defining a correspondence between each of the sites and one or more of the server nodes assigned to the groups containing that site is then provided to a router 12 accessible from the network. The router provides the address of the server node that is to service the next client request for each site. In one group of embodiments, the sites return files in response to the requests, and the measurement of the computational resources includes recording information identifying each returned file, the size of that file, and the number of times that file was returned. Each server node includes a cache memory for facilitating the return of the files in response to the client requests, and the grouping of the sites also depends on the amount of memory in the cache on each of the servers. Sites are partitioned into groups, each of which is assigned to a node, such that the most popular files of all the sites are distributed equally among all the nodes (in other words for each group, the sum of sizes of the popular files included in that group is nearly the same) and additionally, the computational workload impos d on each node is nearly the same. Sites that require more computational resources than can be efficiently

provided by a single server node are replicated on a plurality of server nodes(Page 8, line 25). In the preferred embodiment of the present invention, the groupings are periodically updated by measuring the computational resources required to service the requests to each of the sites over a second time period; and grouping the sites into N new groups(Page 20, line 9). The new groups are constructed by swapping sites between the previous groups. Each new group is assigned to a corresponding one of the server nodes. The groups are constructed such that, for each pair of new groups, the difference in the sum of the measured computational resources over the second time period is within a second predetermined error value. The new grouping that satisfies the second error condition and for which the new groups differ from the previous groups by as few site swaps as possible is the preferred new grouping.

VI. ISSUES

Are Claims 1-12 unpatentable under 35 U.S.C. 103(a) over Yu, US Patent No. 6,351,775 and further in view of Desai, US Patent 6,434,608

VII. GROUPING OF CLAIMS

The claims are to be considered in a plurality of groupings. Claims 1,2, and 8-10 make up the first group. The remaining claims are to be considered separately, i.e., one claim per group.

VIII. ARGUMENT

A. The Examiner's Burden

To sustain a rejection under 35 U.S.C. 103, the Examiner must show that the combined references teach each of the elements of the claim or that there is some motivation in the art for altering the teachings of the references to arrive at the combined set of teachings. "The mere fact that a reference could be modified to produce the patented invention would not make the modification obvious unless it is suggested by the prior art." (*Libbey-Owens-Ford v. BOC Group*, 4 USPQ 2d 1097, 1103). In addition, the Examiner must show that there is some motivation in the art that would cause someone of ordinary skill to combine the references, and that in making the combination, there was a reasonable expectation of success. Where the claimed subject matter has been rejected as obvious in view of a combination of prior art references, a proper analysis under section 103 requires, *inter alia*, consideration of two factors: (1) whether the prior art would have suggested to those of

ordinary skill in the art that they should make the claimed composition or device, or carry out the claimed process; and (2) whether the prior art would also have revealed that in so making or carrying out, those of ordinary skill would have a reasonable expectation of success... Both the suggestion and the reasonable expectation of success must be founded in the prior art, not in the applicant's disclosure. *In re Vaeck*, 20 USPQ2d 1438, 1442(CAFC 1991).

B. Rejections of Claims 1, 2, and 8-10

In making this rejection, the Examiner stated that Yu discloses a method for operating a server cluster comprising N server nodes to service client requests, each client request being directed to one of a plurality of sites hosted on said server cluster, each site being identified by a domain name and each server node being identified by an address on a network connecting said clients to said server nodes. The Examiner points to the abstract of Yu in support of this assertion, (Abstract). Applicant must disagree with the Examiner's reading of Yu.

The Examiner's argument is based on equating a "class" as defined in Yu with a "site" as defined in the present invention. Yu teaches grouping objects specified by URLs into classes. Yu teaches a system in which URLs are grouped into classes and the various classes are assigned to the various servers. Yu defines the term URL as the unique address of information on the Internet (Column 1, line 56-Column 2, line 3). An example of a URL according to Yu is "http://www.philipyu.com:80/table.html". The URL in this example includes the domain name "philipyu.com" as a part thereof; however, a URL, in general, need not include a domain name since not all IP addresses have domain names, e.g. http://186.197.3.35/table.html.

In contrast, the present invention is directed to the grouping of sites on servers. A site is a collection of files associated with a single domain name. Hence, for a site according to the present invention to be equivalent to a class according to Yu, all of the URLs in the class would need to refer to the same domain name. Yu does not teach forming such a class. Yu teaches only forming collections of URLs, referred to as "classes", and making assignments of those classes to servers (column 6, lines 18-29). There is no teaching in Yu that the selected URLs that form a class all refer to the same domain name. In fact, the examples given starting at Column 6, line 18 utilize a hash function to divide the URLs into classes

irrespective of any domain name contained in the URLs. Yu also refers to the use of the PICS labels for dividing URLs into classes by content. Since the PICS labels do not depend on the domain name, this mapping of URLs into classes does not depend on the domain name associated with the URLs. Hence, in general, a class according to Yu will include URLs belonging to a plurality of sites and some that do not belong to any site having a domain name. That is, a class as used by Yu is not identified by "a domain name". In contrast, a site according to the present invention is identified by a single domain name.

In addition, Claim 1 requires the measurement of the computational resources required to service the requests to each of the sites over a first period of time. The Examiner admits that Yu does not teach this limitation. The Examiner looks to Desai for this missing teaching. According to the Examiner, Desai teaches the tracking of requested objects that are retrieved from servers and sent to the requesting host and monitoring how many cache misses have occurred in supplying such objects. The Examiner specifically points to lines 14-44 of column 3. According to the Examiner, an object is equivalent to a computational resource. First, the object being returned is the file specified by the URL; hence, the object is not the computational resource required to supply the object. Second, the cited passage refers to recording the identity of objects for which a request has generated a cache miss in a table. It should be noted that if all of the objects on the server were stored in the cache, the system taught in Desai would not store any information. Hence, the table identified by the Examiner is not a measure of the computational resources needed to service the requests of each of the sites, since only a partial list of the objects is stored. Furthermore, the stored information it is not a measure of the computational resources needed to service a request for a particular object.

In this regard, it should also be noted that there is no correlation between a cache miss and the computational resources needed to deliver the object specified by the URL. A cache miss is generated when the object in question is not in the cache. This occurs the first time the object is requested, regardless of the computational resources required to supply the object. In addition, a miss occurs on subsequent requests for the object if the object has been displaced from the cache by another object. In this case, the miss is a function of the sequence of requests for objects, not the computational resources needed to service th requests for th objects.

The Examiner also points to the passage at column 6, line 62 to column 7, line 15 as supporting the Examiner's contention that Desai teaches measuring computational resources. Applicant disagrees with the Examiner's reading of this passage. The cited passage refers to an algorithm for adjusting a threshold value that determines how may cache misses must occur for an object before the object is cached. The passage refers to keeping track of cache hits per object requested. As noted above, there is no correlation between cache misses, or hits, and the computational resources needed to deliver an object in response to a request for that object.

The Examiner states that one would combine the teachings of Desai into the system of Yu to determine how many cache misses to require before caching an object. Note that this combined system does not satisfy the limitations of Claim 1. The combined system would count the number of cache misses generated by an object that is not currently cached to determine when the object is to be cached on the server to which it is assigned. That is, the resultant system is that of Yu with a more efficient caching algorithm on the individual servers. This is not a system in which the grouping of URLs on the servers is adjusted to keep the difference in the sum of cache misses generated by each site within a predetermined error value. In fact, this system operates independent of the grouping of URLs on the servers.

Claim 1 also requires grouping said sites into N groups, each group being assigned to a corresponding one of said server nodes such that for each pair of groups, the difference in the sum of said measured computational resources is within a first predetermined error value. According to the Examiner, Yu teaches this limitation at col. 6, lines 31-36; col. 7, lines 18-37; and col. 9, lines 27-33. The Examiner refers to the teaching of SA(j) as the number of requests for object classes assigned to server j, but is silent as to how this relates to this contention. Since the Examiner has already admitted that Yu does not teach measuring computational resources, Applicant cannot see how the Examiner has arrived at this conclusion. At most, Yu teaches grouping classes, not sites, utilizing the number of requests C(J,I) for objects in class I on server J. The quantity SA(J) is the sum of C(J,I) over the classes. The Examiner refers to column 6, lines 31-36, column 7, lines 18-37, and column 9, lines 27-33 in support of this assertion. The first passage merely refers to monitoring the total load on each server and changing the class to server assignments to balance the load as

measured by the number of requests for objects. The second passage refers to how a change in assignment is communicated once such a change has been determined to be advantageous. The third passage refers to keeping statistics on the number of requests for objects in class I on server J.

Further, the algorithm taught in Yu moves a class as a whole from one server to another if the servers are unbalanced as measured by the number of requests for each class on each server. It should be noted that Yu measures only the number of requests for each class on each server, not the number of requests for each site on each server. Hence, even if one were to substitute measuring the number of cache misses for each class on each server for C(J,I), and the quantities derived therefrom, i.e., SA(J), the resultant modified version of Yu would not satisfy this limitation of Claim 1. In this regard, it should also be noted that the same argument applies to the limitation of "measuring the computational resources required to service said requests to each of said sites over a first time period", since, at most, the modified version of Yu would not measure any quantity related to each of the sites, no less the computational resources needed. Accordingly, Applicant submits that the Examiner has not made a *primia facia* case for obviousness with respect to Claim 1 or the claims dependent therefrom.

C. Rejection of Claim 3

Claim 3 requires that said sites return files in response to said requests, and that said step of measuring said computational resources comprises recording information identifying each returned file, the size of that file, and the number of times that file was returned. The Examiner maintains that Desai teaches recording information identifying each returned file, the size of that file, and the number of times that file was returned. The Examiner points to Desai, column 3, lines 14-44 and column 6 line 62 to column 7, line 15 in support of this assertion. First, as noted above, Desai only teaches recording the identity of the files for which a cache miss has occurred and the number of cache misses for those files prior to the files being cached. Hence, the recorded information does not identify **each returned file.** Furthermore, the number of cache misses is not a measure of the number of times the file was returned, since once the file is cached, Desai no longer keeps count of the number of times the file is requested. Finally, there is no teaching of recording the file size for these files, no less

all returned files. Hence, Applicant submits that there are additional grounds for allowing Claim 3 and the claims dependent therefrom.

D. Rejection of Claim 4

Claim 4 is dependent from Claim 3 and further requires that each of said server nodes comprises a cache memory for facilitating the return of said files in response to said request and wherein said step of grouping said sites also depends on the amount of memory in said cache memory on each of said servers.

The Examiner maintains that Yu-Desai disclose the limitation that each of said server nodes comprises a cache memory for facilitating the return of said files in response to said request and that said step of grouping said sites also depends on the amount of memory in said cache memory on each of said servers. The Examiner refers to Yu, column 8, line 10 to column 9, line 10 as supporting this contention. The cited passage refers to the manner in which the cache manager on each of the servers decides whether or not an object is to be cached on that server. Similarly, Desai also uses the size of the cache to determine whether or not an object should be cached. The Examiner also points to step 1040 shown in Figure 10 of Yu. Applicant must respectfully point out that the step refers to computing the average number of requests for objects per server. The term "M" refers to the number of servers, not memory. Hence, neither reference teaches grouping objects onto servers based on the amount of memory in the cache on each server, no less grouping sites in this manner. Accordingly, there are additional grounds for allowing Claim 4 and the claims dependent therefrom.

E. Rejection of Claim 5

Claim 5 is dependent from Claim 4 and further requires that said groups are chosen such that said files returned during said first time period more than a predetermined number of times can be stored simultaneously in said cache memory.

The Examiner maintains that Yu-Desai disclose the limitation that said groups are chosen such that said files returned during said first time period more than a predetermined number of times can be stored simultaneously in said cache memory. The Examiner points to Desai, column 3, lines 14-44 in support of this contention. Applicant must disagree. The cited passage refers to the manner in which the system determines whether or not an object is

to be cached on a server. There is no teaching of grouping objects, no less sites, on servers such that all files returned more than a predetermined number of times can be stored simultaneously in the cache memory. Hence, there are additional grounds for allowing Claim 5.

F. Rejection of Claim 6

Claim 6 is dependent from Claim 3 and further requires that said measurement of said computational resources further comprises measuring the number of bytes of data returned in response to said requests for each site during said first time period.

The Examiner stated that Yu-Desai disclose the limitation that said measurement of said computational resources further comprises measuring the number of bytes of data returned in response to said requests for each site during said first time period. The Examiner refers to Desai, column 6, line 62 to column 7, line 10 as supporting this contention. The cited passage refers to using the size of the object relative to an object threshold to determine if the object should be cached in a network cache. There is no teaching of measuring the number of bytes of data returned in response to each request during a first time period. Furthermore, there is no teaching of grouping objects, no less sites, based on such a measurement. The reference uses the information to determine if an object is to be cached, not to determine on which server the object is to be located. Further, once the object is cached, all data collection for that object is suspended. Hence, there are additional grounds for allowing Claim 6 and the claims dependent therefrom.

G. Rejection of Claim 7

Claim 7 is dependent from Claim 6 and further requires estimating the number of bytes of data returned directly from said cache memory in servicing said requests for each site during said first time period.

The Examiner stated that Yu-Desai teach the limitation of estimating the number of bytes of data returned directly from said cache memory in servicing said requests for each site during said first time period. The Examiner points to Desai, column 6, line 62 – column 7, line 10 as supporting this contention. As noted above, the cited passage teaches using the size of a file relative to an object size threshold to decide whether to cache the file. There is no

teaching of estimating the number of bytes returned directly from the cache memory in servicing requests. In this regard, it should be noted that once an object is cached, there is no teaching that the system uses the size of the object for any purpose, no less making such an estimate; hence the system of Desai could not keep track of this information. Accordingly there are additional grounds for allowing Claim 7.

H. Rejection of Claim 11

Claim 11 is dependent from Claim 1 and further requires measuring the computational resources required to service said requests to each of said sites over a second time period; and grouping said sites into N new groups, by swapping sites between said previous groups, each new group being assigned to a corresponding one of said server nodes such that for each pair of new groups, the difference in the sum of said measured computational resources over said second time period is within a second predetermined error value.

The Examiner stated that Yu-Desai disclose measuring the computational resources required to service said requests to each of the sites over a second time period (Desai, col. 3, lines 14-44), and grouping said sites into N new groups, by swapping sites between said previous groups, each new group being assigned to a corresponding one of said server nodes such that for each pair of new groups, the difference in the sum of said measured computational resources over said second time period is within a second predetermined error value (Yu, col. 9, line 51 – col. 10, line 42). Applicant disagrees. As noted above, Yu teaches swapping classes, not sites. Since a site may be split between a number of classes, moving a class does not necessarily imply moving even one site. Accordingly, there are additional grounds for allowing Claim 11 and the claim dependent therefrom.

I. Rejection of Claim 12

Claim 12 is dependent from Claim 11 and further requires that said new groups differ from said previous groups by as few site swaps as possible.

The Examiner stated that Yu-Desai disclose that said new groups differ from said previous groups by as few site swaps as possible (Yu, col. 10, lines 25-42). Applicant disagrees. As noted above, Yu teaches moving classes not sites. Since a class can have a plurality of sites, moving a class does not minimize the number of sites that are moved. For

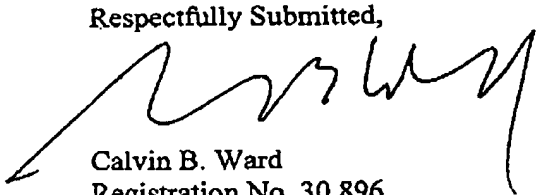
example, if one site has an object that is being requested a large number of times, the minimum move would be to move just the site containing that object. In the scheme taught in Yu, one would move the class containing the object. As noted above, this may involve moving a site, a group of sites, or no site at all. Accordingly, there are additional grounds for allowing Claim 12.

VII. CONCLUSION

Appellants respectfully submit that for the reasons of fact and law argued herein, the decision of the Examiner in finally rejecting Claims 1-12 should be reversed.

I hereby certify that this paper (along with any others attached hereto) is being sent via facsimile to Fax number 703-746-7238

Respectfully Submitted,



Calvin B. Ward
Registration No. 30,896
Date: 8/07/2003

Hewlett-Packard Co.
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400
Telephone (925) 855-0413
Telefax (925)855-9214

APPENDIX

THE CLAIMS ON APPEAL:

1. A method for operating a server cluster comprising N server nodes to service client requests, each client request being directed to one of a plurality of sites hosted on said server cluster, each site being identified by a domain name and each server node being identified by an address on a network connecting said clients to said server nodes, said method comprising the steps of:

measuring the computational resources required to service said requests to each of said sites over a first time period;

grouping said sites into N groups, each group being assigned to a corresponding one of said server nodes such that for each pair of groups, the difference in the sum of said measured computational resources is within a first predetermined error value; and

providing configuration information to a router accessible from said network, said information defining a correspondence between each of said sites and one of said server nodes assigned to one of said groups containing that site, said router providing said address of said server node in response to a message specifying said domain name of said site.

2. The method of Claim 1 wherein said router is a Domain Name System (DNS) server.

3. The method of Claim 1 wherein said sites return files in response to said requests, and wherein said step of measuring said computational resources comprises recording information identifying each returned file, the size of that file, and the number of times that file was returned.

4. The method of Claim 3 wherein each of said server nodes comprises a cache memory for facilitating the return of said files in response to said request and wherein said

step of grouping said sites also depends on the amount of memory in said cache memory on each of said servers.

5. The method of Claim 4 wherein said groups are chosen such that said files returned during said first time period more than a predetermined number of times can be stored simultaneously in said cache memory.

6. The method of Claim 3 wherein said measurement of said computational resources further comprises measuring the number of bytes of data returned in response to said requests for each site during said first time period.

7. The method of Claim 6 further comprising estimating the number of bytes of data returned directly from said cache memory in servicing said requests for each site during said first time period.

8. The method of Claim 1 wherein one of said sites belongs to two of said groups.

9. The method of Claim 1 wherein one of said sites belongs to all of said groups.

10. The method of Claim 1 wherein said router selects which of said service nodes corresponding to said two groups will service a request for that site.

11. The method of Claim 1 further comprising the steps of:

measuring the computational resources required to service said requests to each of said sites over a second time period; and

grouping said sites into N new groups, by swapping sites between said previous groups, each new group being assigned to a corresponding one of said server nodes such that for each pair of new groups, the difference in the sum of said measured computational resources over said second time period is within a second predetermined error value.

12. The method of Claim 11 wherein said new groups differ from said previous groups by as few site swaps as possible.